This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims

1. (Currently Amended) An article of manufacture embodied as one of hardware logic and a computer readable storage medium including program logic for performing configuration checking of a network, wherein the program logic causes operations to be performed, the operations comprising:

scanning a network data store for at least one transaction, wherein said transaction includes one of connecting components in said network, adding components to said network, updating components in said network, and rezoning components in said network;

generating at least one event for said transaction using a mapping of events to transactions;

for said event, obtaining configuration data associated with components in said transaction;

generating at least one trigger for said event, wherein said trigger is associated with at least one configuration policy;

~~associating at least one configuration policy with said event;~~

comparing said configuration policy associated with said trigger with said configuration data associated with said event for which said trigger was generated; and

determining whether said configuration policy has been violated based on the comparison.

2. (Cancelled)

3. (Original) The article of manufacture of claim 1, wherein said configuration policy is retrieved from a local policy data store.

4. (Original) The article of manufacture of claim 3, wherein said configuration policy in the local policy data store is automatically updated with a configuration policy in a remote data store.

5. (Currently Amended) The article of manufacture of claim 1, wherein the operations further comprise:

receiving a hypothetical network scenario, wherein said hypothetical network scenario represents a new network configuration that a system administrator wants to create;

generating at least one transaction based on the hypothetical network scenario;

populating the network data store with configuration data for said transaction; and

after determining whether said configuration policy has been violated based on the comparison, rolling back said transaction.

6. (Original) The article of manufacture of claim 1, wherein the operations further comprise:

receiving a request to perform configuration checking on an existing network configuration.

7. (Original) The article of manufacture of claim 1, wherein the operations further comprise:

when said configuration policy has been violated, performing an action specified in that configuration policy.

8. (Original) The article of manufacture of claim 7, wherein the action is at least one of logging an indication that the configuration policy has been generated, generating at least one policy violation event, sending a notification, and highlighting a network topology viewer that graphically depicts the network.

9. (Original) The article of manufacture of claim 1, wherein the operations further comprise:

when said configuration policy has been violated,

accessing a solution in a knowledge data store; and

applying the solution so that said configuration policy is not violated.

10. (Original) The article of manufacture of claim 1, wherein the operations further comprise:

when said configuration policy has been violated,

determining that a component in the network is able to provide a solution; and

allowing the component to apply the solution so that said configuration policy is not violated.

11. (Currently Amended) The article of manufacture of claim 1, wherein the operations for determining whether said configuration policy has been violated further comprise ~~at least one of~~ identifying incompatibilities between components in the network, performance issues, and availability, wherein said incompatibilities are conflicts between components in said network, said performance relates to whether a desired performance level is met, and said availability relates to whether there is a single point of failure in said network.

12. (Currently Amended) An article of manufacture embodied as one of hardware logic and a computer readable storage medium including program logic for performing proactive configuration checking of a network, wherein the program logic causes operations to be performed, the operations comprising:

receiving a hypothetical network scenario, wherein said hypothetical network scenario represents a new network configuration that a system administrator wants to create;

generating at least one transaction based on said hypothetical network scenario, wherein said transaction includes one of connecting components in said hypothetical network scenario, adding components to said hypothetical network scenario, updating components in said hypothetical network scenario, and rezoning components in said hypothetical network scenario;

populating a network data store with configuration data for said transaction, wherein said configuration data includes configuration data for components in said hypothetical network scenario described by said transaction;

generating at least one event for said transaction using a mapping of events to transactions; and

using configuration data associated with said event to determine whether a configuration policy has been violated.

13. (Original) The article of manufacture of claim 12, wherein the operations further comprise:

rolling back said transaction by removing the configuration data for said transaction from the network data store.

14. (Currently Amended) An article of manufacture embodied as one of hardware logic and a computer readable storage medium including program logic for performing reactive configuration checking of a network, wherein the program logic causes operations to be performed, the operations comprising:

receiving a request to perform configuration checking on an existing network configuration;

scanning a network data store for at least one transaction, wherein said transaction includes one of connecting components in said network, adding components to said network, updating components in said network, and rezoning components in said network;

generating at least one event for said transaction using a mapping of events to transactions; and

using configuration data associated with said event to determine whether a configuration policy has been violated by determining whether the at least one transaction results in incompatibilities, performance issues, and availability issues, wherein said incompatibilities are conflicts between components in the network, said performance issues relate to whether a desired performance level is met, and said availability issues relate to whether there is a single point of failure in the network.

15. (Original) The article of manufacture of claim 14, wherein the operations further comprise:

when said configuration policy has been violated, automatically correcting the violation.

16 - 20. (Cancelled)

21. (Currently Amended) A system for performing configuration checking of a network, comprising:

a processor;

a computer readable storage medium accessible to the processor; and

program logic including code ~~capable of causing~~ that causes the processor to perform:

scanning a network data store for at least one transaction, wherein said transaction includes one of connecting components in said network, adding components to said network, updating components in said network, and rezoning components in said network;

generating at least one event for said transaction using a mapping of events to transactions;

for said event, obtaining configuration data associated with components in said transaction;

generating at least one trigger for said event, wherein said trigger is associated with at least one configuration policy;

~~associating at least one configuration policy with said event;~~

comparing the said configuration policy associated with said trigger with said configuration data associated with said event for which said trigger was generated; and

determining whether said configuration policy has been violated based on the comparison.

22. (Cancelled)

23. (Original) The system of claim 21, wherein said configuration policy is retrieved from a local policy data store.

24. (Original) The system of claim 23,wherein said configuration policy in the local policy data store is automatically updated with a configuration policy in a remote data store.

25. (Currently Amended) The system of claim 21, wherein the code ~~is capable of causing~~ causes the processor to further perform:

receiving a hypothetical network scenario, wherein said hypothetical network scenario represents a new network configuration that a system administrator wants to create;

generating at least one transaction based on the hypothetical network scenario;

populating the network data store with configuration data for said transaction; and

after determining whether said configuration policy has been violated based on the comparison, rolling back said transaction.

26. (Currently Amended) The system of claim 21, wherein the code ~~is capable of causing~~ causes the processor to further perform:

receiving a request to perform configuration checking on an existing network configuration.

27. (Currently Amended) The system of claim 21, wherein the code ~~is capable of causing~~ causes the processor to further perform:

when said configuration policy has been violated, performing an action specified in that configuration policy.

28. (Original) The system of claim 27, wherein the action is at least one of logging an indication that the configuration policy has been generated, generating at least one policy violation event, sending a notification, and highlighting a network topology viewer that graphically depicts the network.

29. (Currently Amended) The system of claim 21, wherein the code ~~is capable of causing~~ causes the processor to further perform:

when said configuration policy has been violated,

accessing a solution in a knowledge data store; and

applying the solution so that said configuration policy is not violated.

30. (Currently Amended) The system of claim 21, wherein the code ~~is capable of causing~~ causes the processor to further perform:

when said configuration policy has been violated,

determining that a component in the network is able to provide a solution; and

allowing the component to apply the solution so that said configuration policy is not violated.

31. (Currently Amended) The system of claim 21, wherein the code for determining whether said configuration policy has been violated ~~is capable of causing~~ causes the processor to further perform ~~at least one of~~ identifying incompatibilities between components in the network, performance issues, and availability issues, wherein said incompatibilities are conflicts between components in said network, said performance relates to whether a desired performance level is met, and said availability relates to whether there is a single point of failure in said network.

32. (Currently Amended) A system for performing proactive configuration checking of a network, comprising:

a processor;

a computer readable storage medium accessible to the processor; and

program logic including code ~~capable of causing~~ that causes the processor to perform:

receiving a hypothetical network scenario, wherein said hypothetical network scenario represents a new network configuration that a system administrator wants to create;

generating at least one transaction based on said hypothetical network scenario, wherein said transaction includes one of connecting components in said hypothetical network scenario, adding components to said hypothetical network scenario, updating components in said hypothetical network scenario, and rezoning components in said hypothetical network scenario;

populating a network data store with configuration data for said transaction, wherein said configuration data includes configuration data for components in said hypothetical network scenario described by said transaction;

generating at least one event for said transaction using a mapping of events to transactions; and

using configuration data associated with said event to determine whether a configuration policy has been violated.

33. (Currently Amended) The system of claim 32, wherein the code ~~is capable of causing~~ causes the processor to further perform:

rolling back said transaction by removing the configuration data for said transaction from the network data store.

34. (Currently Amended) A system for performing reactive configuration checking of a network, comprising:

a processor;

a computer readable storage medium accessible to the processor; and

program logic including code ~~capable of causing~~ that causes the processor to perform:

receiving a request to perform configuration checking on an existing network configuration;

scanning a network data store for at least one transaction, wherein said transaction includes one of connecting components in said network, adding components to said network, updating components in said network, and rezoning components in said network;

generating at least one event for said transaction using a mapping of events to transactions; and

using configuration data associated with said event to determine whether a configuration policy has been violated by determining whether the at least one transaction results in incompatibilities, performance issues, and availability issues, wherein said incompatibilities are conflicts between components in the network, said performance issues relate to whether a desired performance level is met, and said availability issues relate to whether there is a single point of failure in the network.

35. (Currently Amended) The system of claim 34, wherein the code ~~is capable of causing~~ causes the processor to further perform:

when said configuration policy has been violated, automatically correcting the violation.

36-40. (Cancelled)

41. (Currently Amended) A method for performing configuration checking of a network, comprising:

scanning a network data store for at least one transaction, wherein said transaction includes one of connecting components in said network, adding components to said network, updating components in said network, and rezoning components in said network;

generating at least one event for said transaction using a mapping of events to transactions;

for said event, obtaining configuration data associated with components in said transaction;

generating at least one trigger for said event, wherein said trigger is associated with at least one configuration policy;

~~associating at least one configuration policy with said event;~~

comparing the said configuration policy associated with said trigger with said configuration data associated with said event for which said trigger was generated; and

determining whether said configuration policy has been violated based on the comparison.

42. (Currently Amended) A method for performing proactive configuration checking of a network, comprising:

receiving a hypothetical network scenario, wherein said hypothetical network scenario represents a new network configuration that a system administrator wants to create;

generating at least one transaction based on said hypothetical network scenario, wherein said transaction includes one of connecting components in said hypothetical network scenario, adding components to said hypothetical network scenario, updating components in said hypothetical network scenario, and rezoning components in said hypothetical network scenario;

populating a network data store with configuration data for said transaction, wherein said configuration data includes configuration data for components in said hypothetical network scenario described by said transaction;

generating at least one event for said transaction using a mapping of events to transactions; and

using configuration data associated with said event to determine whether a configuration policy has been violated.

43. (Currently Amended) A method for performing reactive configuration checking of a network, comprising:

receiving a request to perform configuration checking on an existing network configuration;

scanning a network data store for at least one transaction, wherein said transaction includes one of connecting components in said network, adding components to said network, updating components in said network, and rezoning components in said network;

generating at least one event for said transaction using a mapping of events to transactions; and

using configuration data associated with said event to determine whether a configuration policy has been violated by determining whether the at least one transaction results in incompatibilities, performance issues, and availability issues, wherein said incompatibilities are conflicts between components in the network, said performance issues relate to whether a desired performance level is met, and said availability issues relate to whether there is a single point of failure in the network.

44. (Cancelled)

45. (New) The method of claim 41, wherein said configuration policy is retrieved from a local policy data store.

46. (New) The method of claim 45, wherein said configuration policy in the local policy data store is automatically updated with a configuration policy in a remote data store.

47. (New) The method of claim 41, further comprising:

receiving a hypothetical network scenario, wherein said hypothetical network scenario represents a new network configuration that a system administrator wants to create;

generating at least one transaction based on the hypothetical network scenario;

populating the network data store with configuration data for said transaction; and

after determining whether said configuration policy has been violated based on the comparison, rolling back said transaction.

48. (New) The method of claim 41, further comprising:

receiving a request to perform configuration checking on an existing network configuration.

49. (New) The method of claim 41, further comprising:

when said configuration policy has been violated, performing an action specified in that configuration policy.

50. (New) The method of claim 49, wherein the action is at least one of logging an indication that the configuration policy has been generated, generating at least one policy violation event, sending a notification, and highlighting a network topology viewer that graphically depicts the network.

51. (New) The method of claim 41, further comprising:

when said configuration policy has been violated,

accessing a solution in a knowledge data store; and

applying the solution so that said configuration policy is not violated.

52. (New) The method of claim 41, further comprising:

when said configuration policy has been violated,

determining that a component in the network is able to provide a solution; and

allowing the component to apply the solution so that said configuration policy is not violated.

53. (New) The method of claim 41, wherein determining whether said configuration policy has been violated further comprises identifying incompatibilities between components in the network, performance issues, and availability, wherein said incompatibilities are conflicts

between components in said network, said performance relates to whether a desired performance level is met, and said availability relates to whether there is a single point of failure in said network.

54. (New) The method of claim 42, further comprising:

rolling back said transaction by removing the configuration data for said transaction from the network data store.

55. (New) The method of claim 43, further comprising:

when said configuration policy has been violated, automatically correcting the violation.